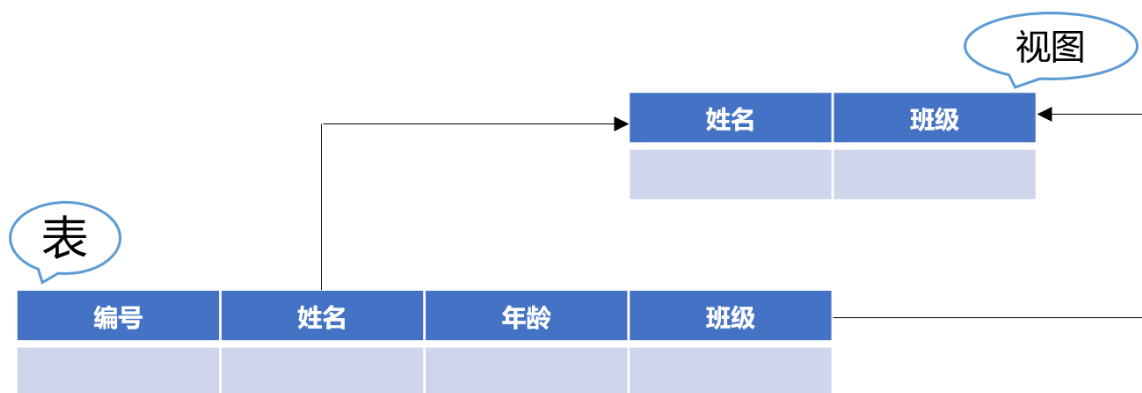


数据库的基础知识

三级模式：

1. **外模式**：面向应用程序，描述用户的数据视图（View）。

可以为不同用户的需求创建不同的视图。



基本表中的数据是实际存储在数据库中的。

视图中的数据是查询或计算出来的。

不同用户的需求不同，数据的显示方式也会多种多样。

一个数据库中会有多个外模式，而概念模式和内模式则只有一个。

2. **概念模式**：又称为模式、逻辑模式，面向数据库设计人员，描述数据的整体逻辑结构。

类似于表格的列标题，它描述了商品表中包含哪些信息。

编号	姓名	年龄	班级

表格的横向称为行，纵向称为列，第一行就是列标题，用来描述该列的数据表示什么含义。

3. **内模式**：又称为物理模式、存储模式，面向物理上的数据库，描述数据在磁盘中如何存储。

在将Excel表格另存为文件时，可以选择保存的文件路径、保存类型（如xls、xlsx、csv等格式）等，这些与存储相关的描述信息相当于内模式。

在数据库中，内模式描述数据的物理结构和存储方式，如堆文件、索引文件、散列（Hash）文件等。

数据模型：

1. **实体** (Entity) 是指客观存在并可相互区分的事物。

例如，学生、书籍、手机都是实体。

2. **属性** (Attribute) 是指实体所具有的某一特性，一个实体可由若干个属性来描述。

例如，书籍实体的属性有ISBN号、书名和页数。

属性由两部分组成，分别是属性名和属性值。

例如，ISBN号和书名是属性名，而“123456、MySQL基础”这些具体值是属性值。

3. **联系**：是指实体与实体之间的联系，有一对一、一对多、多对多三种情况。

例如，每本书都有一个ISBN号，书和书的电子版之间是一一对一的联系；一个作者可以写很多本书，作者和书是一对多的联系；一本书被多个读者阅读，一个读者又可以阅读多本书，读者和书籍之间就形成了多对多的联系。

4. **E-R图**：也称为实体-联系图。

E-R图是一种用图形表示的实体联系模型。

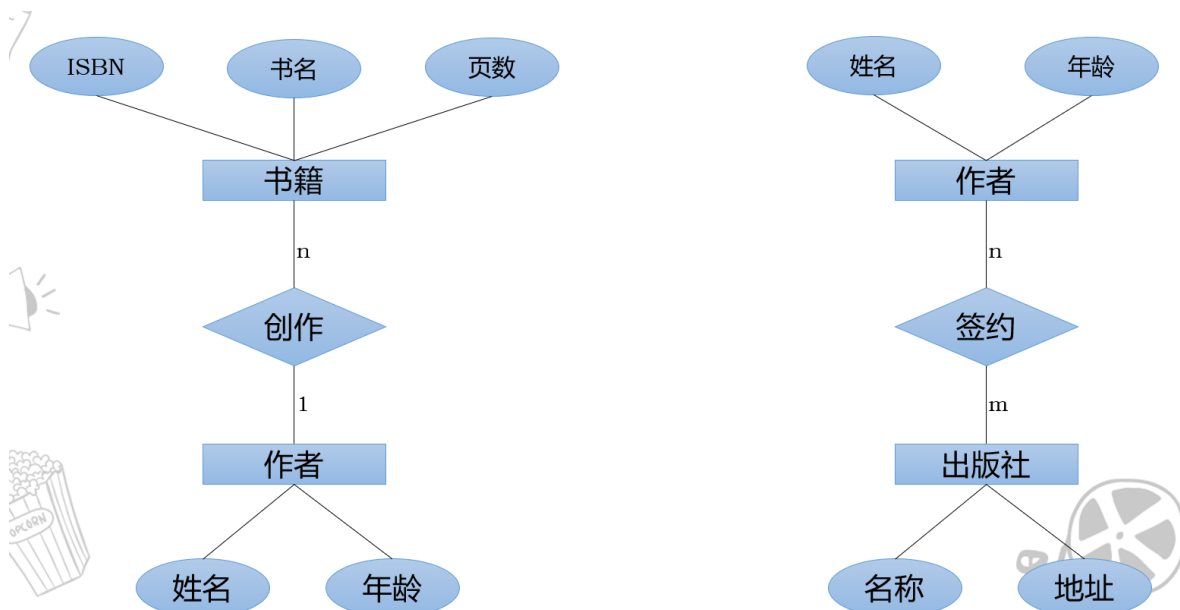
E-R图提供了表示实体型、属性和联系的方法，用来描述现实世界的概念模型。

E-R图通用的表示方式如下。

实体：用矩形框表示，将实体名写在框内。

属性：用椭圆框表示，将属性名写在框内，用连线将实体与属性连接。

联系：用菱形框表示，将联系名写在框内，用连线将相关的实体连接，并在连线旁标注联系类型（一对一“1:1”、一对多“1:n”、多对多“n:m”）。



5. 属性

属性（字段）

ISBN号	书名	页数
123456	MySQL基础	256
456789	深度学习基础	512

元组（记录）

属性：二维表中的列称为属性，每个属性都有一个属性名。

元组：二维表中的每一行数据称为一个元组。

域：域是指属性的取值范围，例如，性别属性的域为男、女。

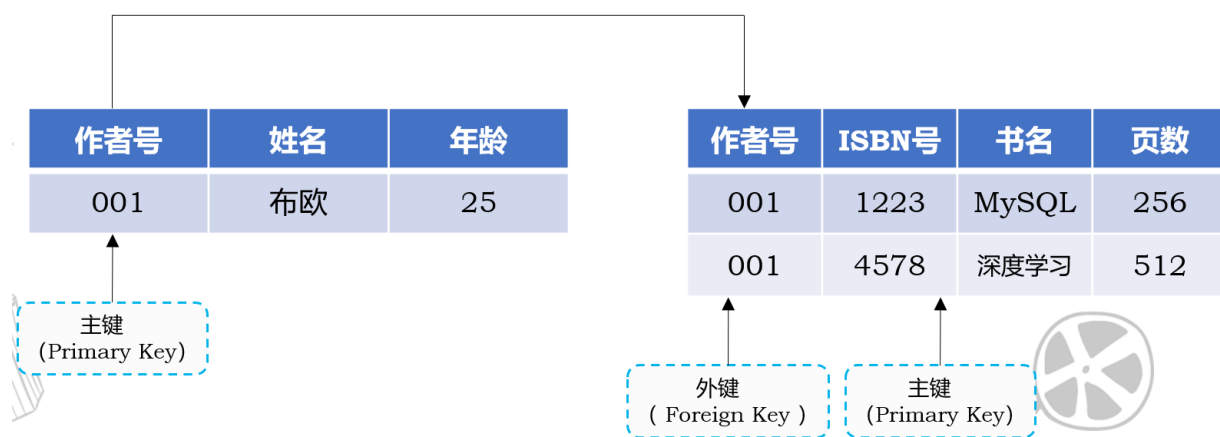
6. **关系模式：**是关系的描述，通常可以简记为：关系名（属性1，属性2，...，属性n）”。

例如，书籍（ISBN号，书名，页数）。

7. **键（Key）：**在二维表中，唯一标识某一条记录，又称为关键字、码。

例如，书籍的ISBN号具有唯一性，ISBN号可以作为书籍实体的键。而书名、页数可能存在重复，不适合作为键。

通过键可以为两张表建立联系（一对多）



多对多联系的表可以转换成两个一对多的表



8. **关系模型的完整性**：为了保证数据库中数据的正确性和相容性，需要对关系模型进行完整性约束。完整性通常包括实体完整性、参照完整性和用户自定义完整性。

9. **实体完整性**：要求关系中的主键不能重复，且不能取空值。空值是指不知道、不存在或无意义的值。

10. **参照完整性**：要求关系中的外键要么取空值，要么取被参照关系中的某个元组的主键值。

11. **用户自定义完整性**：是用户针对具体的应用环境定义的完整性约束条件，由DBMS检查用户自定义的完整性。

关系运算：

1. 关系代数运算符

关系运算符	含义	关系运算符	含义
\cup	并	σ	选择
$-$	差	π	投影
\cap	交	\bowtie	连接
\times	笛卡尔积	\div	除

并 (Union) 、差 (Difference) 、交 (Intersection) 设有关系R和关系S

$R \cup S$ 表示合并两个关系中的元组，数学描述： $R \cup S = \{t \mid t \in R \vee t \in S\}$

$R - S$ 表示找出属于R但不属于S的元组，数学描述： $R - S = \{t \mid t \in R \wedge t \notin S\}$

$R \cap S$ 表示找出既属于R又属于S的元组，数学描述： $R \cap S = \{t \mid t \in R \wedge t \in S\}$

\vee 表示逻辑运算符或， \wedge 表示逻辑运算符与

R		S		$R \cup S$	
学号	学生姓名	学号	学生姓名	学号	学生姓名
1	张三	1	张三	1	张三
2	李四	3	小明	2	李四
$R - S$		$R \cap S$		3	小明
学号	学生姓名	学号	学生姓名		
2	李四	1	张三		

笛卡尔积 (Cartesian Product) 设关系R有n个属性，关系S有m个属性

R和S的笛卡尔积：元组的前n个属性来自R，后m个属性来自S。

结果属性个数= $n+m$ ，结果元组的总个数=R和S中的元组的乘积。

R		$R \times S$			
学号	学生姓名	学号	学生姓名	班级号	班级名称
1	张三	1	张三	1	软件班
2	李四	1	张三	2	设计班
S		2	李四	1	软件班
班级号	班级名称	2	李四	2	设计班
1	软件班				
2	设计班				

选择 (Selection) 、投影 (Projection)

选择操作： $\sigma_{\text{学号}=1}(R)$ ，表示在关系R中查找学号为1的学生。

投影操作： $\pi_{\text{学号}, \text{学生姓名}}(R)$ ，表示在关系R中查找学号和学生姓名。

R			$\sigma_{\text{学号}=1}(R)$			$\pi_{\text{学号}, \text{学生姓名}}(R)$	
学号	学生姓名	学生性别	学号	学生姓名	学生性别	学号	学生姓名
1	张三	男	1	张三	男	1	张三
2	李四	女				2	李四

连接 (Join) :

等值连接：是在R和S的笛卡尔积中选取A、B属性值相等的元组。

自然连接：是一种特殊的等值连接，要求R和S必须有相同的属性组，进行等值连接后再去除重复的属性组。



除 (Division) :

R是学生选课表。

$R \div S1$ 表示查询学号为2的学生所选的课程。

$R \div S2$ 表示查询学号为2和3的学生共同选择的课程。

R		S1	S2
课程号	学号	学号	学号
1	2	2	2
2	2		3
3	2		
1	3		
2	3		
1	4		

$R \div S1$

课程号
1
2
3

$R \div S2$

课程号
1
2

SQL语言:

1. SQL的组成

- 1. 数据定义语言 (Data Definition Language, DDL) 主要用于定义数据库、表等。例如，CREATE语句、ALTER语句、DROP语句等。
- 2. 数据操作语言 (Data Manipulation Language, DML) 主要用于对数据库进行添加、修改和删除操作（增删改）。

例如，INSERT语句、UPDATE语句、DELETE语句。

3. 数据查询语言（Data Query Language, DQL）主要用于查询数据。

例如，使用SELECT语句可以查询数据库中的一条数据或多条数据。

4. 数据控制语言（Data Control Language, DCL）主要用于控制用户的访问权限。

例如，GRANT语句、REVOKE语句、COMMIT语句、ROLLBACK语句。

需要环境：

1. 操作系统：Windows11

2. 系统内存：>=8G

3. 数据库：MySQL8.0

<https://www.mysql.com>

4. 图形化工具：Navicat、VsCode